



₩ 第

### Item 1/30

Which of the following sentences is true?

nums = [1, 2, 3]
vals = nums

vals is longer than nums

nums is longer than vals

nums and vals are different lists

nums and vals are different names of the same

list

Next →

**Retake Test** 



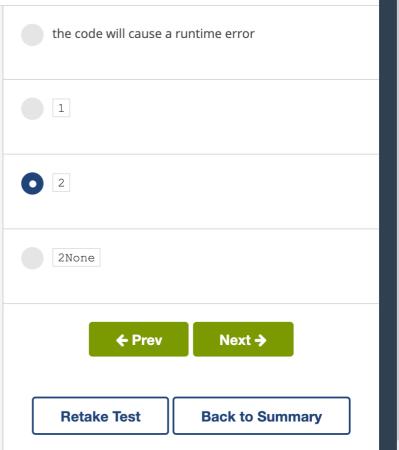


**派** 

## Item 2/30

What is the output of the following snippet?

```
def fun(x):
    if x % 2 == 0:
        return 1
    else:
        return 2
print(fun(fun(2)))
```







## Item 3/30



What is the output of the following piece of code?











**Retake Test** 



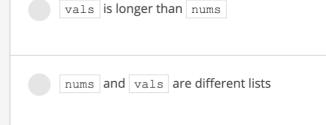


**派** 

# Item 4/30

Take a look at the snippet and choose the <u>true</u> statement:

```
nums = [1, 2, 3]
vals = nums
del vals[:]
```



- nums and vals are different names of the same
- the snippet will cause a runtime error
  - nums is longer than vals



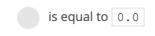


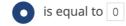
₩ 第

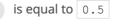
## Item 5/30

The result of the following division:

1 // 2







cannot be predicted



Next →

**Retake Test** 





**永** 

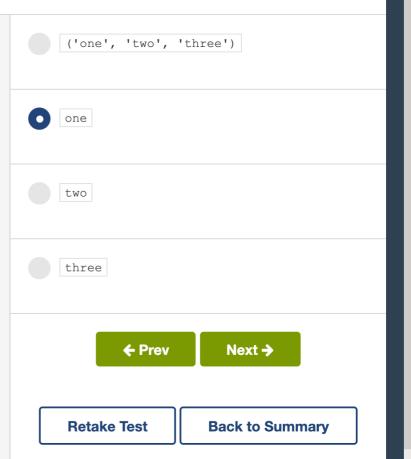
## Item 6/30

What is the output of the following snippet?

```
dct = { 'one':'two', 'three':'one', 'two':
    v = dct['three']

for k in range(len(dct)):
    v = dct[v]

print(v)
```



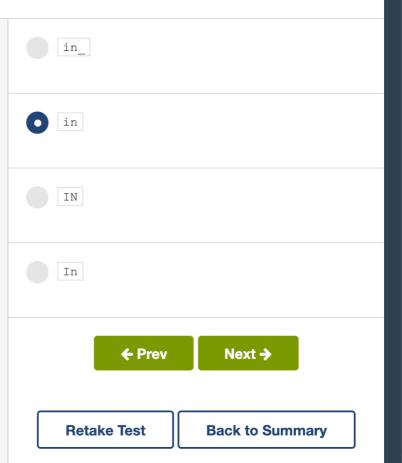




### Item 7/30



One of the following variable names is <u>illegal</u> - which one?







## Item 8/30



What is the output of the following snippet?

```
def fun(x, y):
    if x == y:
        return x
    else:
        return fun(x, y-1)

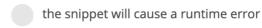
print(fun(0, 3))
```













Next →

**Retake Test** 





₩ 第

## Item 9/30

What is the output of the following snippet?

```
def fun(inp=2, out=3):
    return inp * out
print(fun(out=2))
```



the snippet is erroneous







Next →

**Retake Test** 





**派** 

## Item 10/30

Assuming that the tuple is a correctly created tuple, the fact that tuples are immutable means that the following instruction:

tuple[1] = tuple[1] + tuple[0]

- is illegal
- may be illegal if the tuple contains strings
  - is fully correct
- can be executed if and only if the tuple contains at least two elements

← Prev Next →

**Retake Test** 





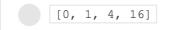
### Item 11/30

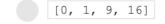
## **派**

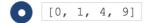
What is the output of the following snippet?

```
list = [x * x for x in range(5)]
def fun(lst):
    del lst[lst[2]]
    return lst

print(fun(list))
```











Next →

**Retake Test** 





## Item 12/30



What is the output of the following piece of code?

$$x = 1 // 5 + 1 / 5$$
  
print(x)











0.4



Next →

**Retake Test** 



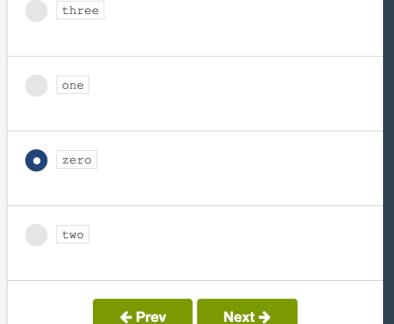


₩ 第

## Item 13/30

How many elements does the lst list contain?

lst = [i for i in range(-1, -2)]



**Retake Test** 





## Item 14/30



What is the output of the following snippet?

```
dct = {}
dct['1'] = (1, 2)
dct['2'] = (2, 1)

for x in dct.keys():
    print(dct[x][1],end="")
```











Next →

**Retake Test** 



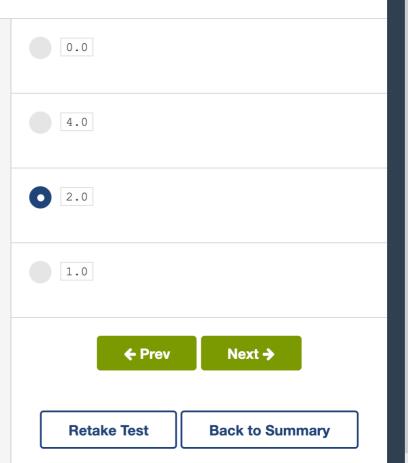


**派** 

## Item 15/30

What is the output of the following piece of code if the user enters two lines containing 2 and 4 respectively?

```
x = float(input())
y = float(input())
print(y ** (1 / x))
```







## Item 16/30



How many hashes (#) will the following snippet send to the console?

```
lst = [[x for x in range(3)] for y in range(3)]
for r in range(3):
    for c in range(3):
        if lst[r][c] % 2 != 0:
            print("#")
```



six

nine

three

zero



Next →

**Retake Test** 

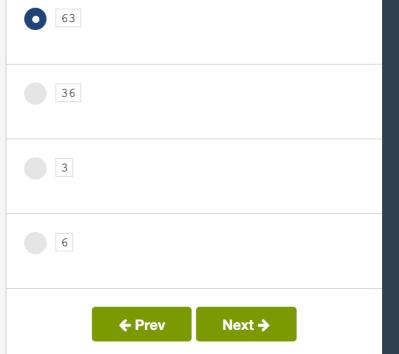




₩ 第

## Item 17/30

What is the output of the following piece of code if the user enters two lines containing 3 and 6 respectively?



**Back to Summary** 

**Retake Test** 





₩ 第

## Item 18/30

#### The following snippet:

def func(a, b):
 return b \*\* a

print(func(b=2, 2))











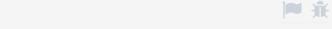
Next →

Retake Test





## Item 19/30



What is the output of the following piece of code?

print("a", "b", "c", sep="sep")



abc



a b c





Next →

**Retake Test** 





## Item 20/30



What value will be assigned to the x variable?

$$z = 0$$

$$y = 10$$

$$x = y < z$$
 and  $z > y$  or  $y > z$  and  $z < y$ 













Next →

**Retake Test** 





## Item 21/30



The meaning of a *positional argument* is determined by:





its connection with existing variables

the argument's name specified along with its value



Next →

**Retake Test** 





### Item 22/30



**永** 

What is the output of the following snippet?









(4)



44



Next →

**Retake Test** 





₩ 第

## Item 23/30

### The following snippet:

def func1(a):
 return None

def func2(a):
 return func1(a) \* func1(a)

print(func2(2))

will output 2

will output 16

will cause a runtime error

will output 4

← Prev

Next →

**Retake Test** 





**派** 

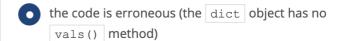
## Item 24/30

What is the output of the following snippet?

```
dd = { "1":"0", "0":"1" }
for x in dd.vals():
    print(x, end="")
```











**Retake Test** 





## Item 25/30



How many stars (  $\star$  ) will the following snippet send to the console?

```
i = 0
while i < i + 2:
    i += 1
    print("*")
else:
    print("*")</pre>
```



two

two

one

 the snippet will enter an infinite loop, printing one star per line

← Prev

Next →

**Retake Test** 





## Item 26/30



What will be the output of the following snippet?











Next →

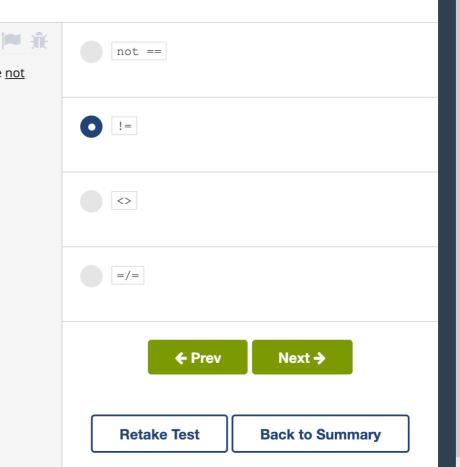
**Retake Test** 





## Item 27/30

An operator able to check whether two values are <u>not</u> <u>equal</u> is coded as:





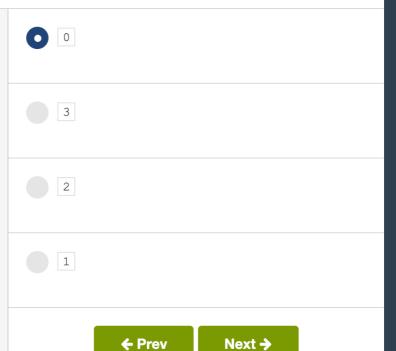


## Item 28/30



What is the output of the following piece of code if the user enters two lines containing 3 and 2 respectively?

```
x = int(input())
y = int(input())
x = x % y
x = x % y
y = y % x
print(y)
```



**Back to Summary** 

**Retake Test** 

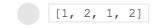


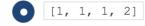


## Item 29/30



What is the output of the following snippet?











Next →

**Retake Test** 





## Item 30/30

● 流

Which of the following lines incorrectly invokes the

def fun(a, b, c=0):

function defined as:





← Prev

**Retake Test**